

Amendments to the Claims

Please amend claims 1, 3, 4, 12, 13, 21, 24-26, 28 and 29. Please cancel claims 2, 12, 22 and 23. Please add new claim 30. The currently pending claims after amendment are listed below.

1. (Currently Amended) A method of testing a device driver on a data processing system, said method comprising:
 - allocating, by an operating system, a data space for executing a device driver;
 - setting up, by the operating system, a data exception handler that emulates a target device;
 - executing the device driver as an application on top of the operating system to test the device driver;
 - monitoring to detect whether a request made by the device driver specifies a target address within the data space; and
 - in response to detecting the target address for the request being made outside of the data space, trapping on the target address for the request and executing a said data exception handler that emulates a target device.
- wherein the setting up step further comprises:
 - (a) calling the operating system to install a data exception handler facility containing the data exception handler into a vector of an interrupt table for the operating system; and
 - (b) calling the data exception handler facility to register data that is used to determine when the data exception is to be taken in a data exception database table.
2. (Cancelled)

1 3. (Currently Amended) ~~The A method according to Claim 2 of testing a device driver on a~~
2 data processing system, said method comprising:

3 allocating, by an operating system, a data space for executing a device driver;

4 setting up, by the operating system, a data exception handler that emulates a target device;

5 executing the device driver as an application on top of the operating system to test the
6 device driver;

7 monitoring to detect whether a request made by the device driver specifies a target address
8 within the data space; and

9 in response to detecting the target address for the request being made outside of the data
10 space, trapping on the target address for the request and executing said data exception handler that
11 emulates a target device

12 wherein the setting up step further comprises:

13 (a) calling the operating system to install a data exception handler facility containing the
14 data exception handler into a vector of an interrupt table for the operating system; and

15 (b) calling the data exception handler facility to register data that is used to determine
16 when the data exception is to be taken wherein the data include each of various target
17 addresses at which the data exception is to be taken, lengths of the target addresses, and
18 user callback routines for calling back to the application into a data exception database
19 table.

1 4. (Currently Amended) The method according to Claim 3, further ~~comprises~~ comprising:
2 passing, by the data exception handler facility, the data space to the vector of the interrupt
3 table.

1 5. (Original) The method according to Claim 4, wherein the data exception handler is a
2 memory mapped input/output (IO) exception handler comprising a parser with a disassembler that
3 is used for disassembling and identifying an assembler instruction of the request.

1 6. (Original) The method according to Claim 5, wherein the assembler instruction is
2 identified as a load command.

1 7. (Original) The method according to Claim 5, wherein the assembler instruction is
2 identified as a store command.

1 8. (Original) The method according to Claim 3, wherein the data exception handler further
2 comprises:
3 saving, into a memory stack, data for the request which includes at least the target address
4 for the request, content data relating to the request, and a data address range for the content data;
5 determining whether the target address that is saved into the memory stack is within the
6 database exception database table;
7 immediately terminating the method of testing if the target address is not within the
8 database exception database table;
9 disassembling the data for the request into disassembled information and passing the
10 disassembled information into a respective one of the user callback routines stored in the data
11 exception database table wherein the respective one of the user callback routines that is used is
12 based on the request and is for emulating a behavior of the target device;
13 emulating, by the respective one of the user callback routines, the behavior of the target
14 device;
15 setting a next instruction address to an address in the memory stack that is after the
16 currently saved target address; and
17 unwinding the memory stack by the data exception handler to return control back to the
18 application.

1 9. (Original) The method according to Claim 8, further comprising:
2 copying, as necessary, the content data by the respective one of the user callback routines
3 from another data space for the respective one of the user callback routines based on the request.

1 10. (Original) The method according to Claim 1, wherein the data exception handler is a
2 software emulator of the target device and further comprises:
3 using a hardware emulator to test the application for the device driver if the hardware
4 emulator is coupled to the data processing system; and
5 using the software emulator to test the application for the device driver if the hardware
6 emulator is not coupled to the data processing system.

1 11. (Original) The method according to Claim 1, further comprises:
2 determining whether the application for the device driver has finished executing;
3 continuing with the executing of the application if the application has not finished
4 executing; and
5 terminating the method of testing when the application has finished executing.

12. (Cancelled)

1 13. (Currently Amended) ~~The~~ A data processing system according to Claim 12 for testing a
2 device driver, said data processing system comprising:
3 a processor and a memory system, wherein:
4 said processor executes an operating system that allocates a data space for executing
5 applications and executes a device driver as an application on top of the operating system to test
6 the device driver;

1 said processor and said memory system, responsive to detecting a request by the device
2 driver specifying a target address outside of the data space, trap on the target address and execute
3 a data exception handler that emulates a target device of the device driver; and

4 said processor and said memory system call a data exception handler facility to register
5 data that is used to determine when a data exception is to be taken, wherein the data include each
6 of various target addresses at which the data exception is to be taken, lengths of the target
7 addresses, and user callback routines for calling back to the application into a data exception
8 database table.

1 14. (Original) The system according to Claim 13, wherein said processor and said memory
2 system call the operating system to install the data exception handler facility containing the data
3 exception handler into a vector of an interrupt table for the operating system.

1 15. (Original) The system according to Claim 14, wherein the data exception handler facility
2 passes the data space to the vector of the interrupt table.

1 16. (Original) The system according to Claim 15, wherein the data exception handler is a
2 memory mapped input/output (IO) exception handler that is stored in the memory system and
3 comprises a parser with a disassembler that is used for disassembling and identifying an
4 assembler instruction of the request.

1 17. (Original) The system according to Claim 16, wherein the assembler instruction is a load
2 command.

1 18. (Original) The system according to Claim 16, wherein the assembler instruction is a store
2 command.

1 19. (Original) The system according to Claim 14, wherein the data exception handler
2 determines whether the target address is within the data exception database table and, responsive
3 to said determination, immediately terminates testing of the device driver if the target address is
4 not within the data exception database table, and, emulates, by the respective one of the user
5 callback routines, the behavior of the target device if the target address is within the data
6 exception database table.

1 20. (Original) The system according to Claim 19, wherein the data exception handler further:
2 copies, as necessary, the content data by the respective one of the user callback routines
3 from another data space for the respective one of the user callback routines based on the request.

1 21. (Currently Amended) The system according to ~~Claim 12~~ Claim 13, wherein the data
2 exception handler is a software emulator of the target device and wherein:
3 a hardware emulator is used to test the application for the device driver if the hardware
4 emulator is coupled to the data processing system; and
5 the software emulator is used to test the application for the device driver if the hardware
6 emulator is not coupled to the data processing system.

22-23 (Cancelled)

1 24. (Currently Amended) ~~The~~ A program product ~~according to Claim 23,~~ for testing a device
2 driver on a data processing system comprising:

3 instruction means for allocating, by an operating system, a data space for executing a
4 device driver;

5 instruction means for setting up, by the operating system, a data exception handler that
6 emulates a target device;

7 instruction means for executing the device driver as an application on top of the operating
8 system to test the device driver;

9 instruction means for monitoring to detect whether a request made by the device driver
10 specifies a target address within the data space;

11 in response to detecting the target address for the request being made outside of the data
12 space, instruction means for trapping on the target address for the request and executing said data
13 exception handler that emulates a target device; and

14 computer usable media bearing said instruction means;

15 wherein the instruction means for setting up further comprises:

16 (a) instruction means for calling the operating system to install a data exception handler
17 facility containing the data exception handler into a vector of an interrupt table for the
18 operating system; and

19 (b) instruction means for calling the data exception handler facility to register data that is
20 used to determine when the data exception is to be taken wherein the data include each of
21 various target addresses at which the data exception is to be taken, lengths of the target
22 addresses, and user callback routines for calling back to the application into a data
23 exception database table.

1 25. (Currently Amended) The program product according to Claim 24, further ~~comprises~~
2 comprising:

3 instruction means for passing, by the data exception handler facility, the data space to the
4 vector of the interrupt table.

1 26. (Currently Amended) The program product according to ~~Claim 22~~ Claim 24, wherein the
2 data exception handler further comprises:

3 instruction means for saving, into a memory stack, data for the request which includes at
4 least the target address for the request, content data relating to the request, and a data address
5 range for the content data;

6 instruction means for determining whether the target address that is saved into the memory
7 stack is within the database exception database table;

8 instruction means for immediately terminating the testing of the device driver if the target
9 address is not within the database exception database table;

10 instruction means for disassembling the data for the request into disassembled information
11 and passing the disassembled information into a respective one of the user callback routines
12 stored in the data exception database table wherein the respective one of the user callback routines
13 that is used is based on the request and is for emulating a behavior of the target device;

14 instruction means for emulating, by the respective one of the user callback routines, the
15 behavior of the target device;

16 instruction means for setting a next instruction address to an address in the memory stack
17 that is after the currently saved target address; and

18 instruction means for unwinding the memory stack by the data exception handler to return
19 control back to the application.

1 27. (Original) The program product according to Claim 26, further comprising:
2 instruction means for copying, as necessary, the content data by the respective one of the
3 user callback routines from another data space for the respective one of the user callback routines
4 based on the request.

1 28. (Currently Amended) The program product according to ~~Claim 22~~ Claim 24, wherein the
2 data exception handler is a software emulator of the target device and further comprises:
3 instruction means for using a hardware emulator to test the application for the device driver
4 if the hardware emulator is coupled to the data processing system; and
5 instruction means for using the software emulator to test the application for the device
6 driver if the hardware emulator is not coupled to the data processing system.

1 29. (Currently Amended) The program product according to ~~Claim 22~~ Claim 24, further
2 ~~comprises~~ comprising:
3 instruction means for determining whether the application for the device driver has finished
4 executing;
5 instruction means for continuing with the executing of the application if the application has
6 not finished executing; and
7 instruction means for terminating the testing of the device driver when the application has
8 finished executing.

1 30. (New) The method according to Claim 1, wherein said data registered in said data
2 exception database table that is used to determine when the data exception is to be taken includes
3 at least one target address at which the data exception is to be taken and at least one user callback
4 routine for calling back to the application.